What is this page?

html tree pages [Search] powered by Google

Search WWW Search aec.cadinfo.net

Web

Results **101 - 110** of about **1,620,000** for **html tree pages**. (**0.53** seconds)

powered by Google

[PDF] <a href="http://www.ucpress.edu/books/**pages**/9670.**html**">buy this ...
File Format: PDF/Adobe Acrobat - View as HTML
Diagnosis of **tree** health problems involves identifying the cause. from the symptoms (expressions of damage to the **tree**), signs ...
www.ucpress.edu/books/**pages**/9670/9670.sample.pdf - Similar pages

Trees: A Treemendous Renewable Resource
Trees are also very important for us as a renewable resource. **Trees** are a natural resource that can be renewed - by the planting of **trees** - replacing the ...
www.kidsface.org/**pages**/resource.**html** - Similar pages

search.cpan.org: **HTML**::**Tree**::Scanning -- article: "Scanning **HTML**"
Now, it may happen that you're rendering that whole **HTML tree** into some layout ... Or you could be trying to make some systematic change to the **HTML tree** ...
search.cpan.org/~sburke/HTML-Tree-3.18/ lib/HTML/Tree/Scanning.pod - Similar pages

Lake Mead NRA Official Expanded Web **Pages**
___ Cercidium floridum Gray ssp. floridum Blue paloverde **Tree**. ALIEN. ... ___ Robinia neomexicana Gray New Mexico locust Shrub or small **tree**. Parashant NM. ...
www.nps.gov/lame/plants.**html** - Similar pages

JJ O'Donnell: St. Augustine to NREN
St. Augustine to NREN:; The **tree** of knowledge and how it grows * ... Then in the front of the manuscript are **pages** in which architectural ornament ...
ccat.sas.upenn.edu/jod/nasig.**html** - Similar pages

Ailanthus altissima (**Tree**-of-Heaven)
Success Story: Cosumnes, **HTML** · Adobe Acrobat · MS Word. Recent comments, **HTML** · Adobe Acrobat ... A few young **trees**, Audacious **tree**, A full **tree**. Bold **tree** ...
tncweeds.ucdavis.edu/esadocs/ailaalti.**html** - Similar pages

The TEX Live Guide TeX Collection 2005
Tree for the programs, along with their support files and documentation. ... The documentation for the programs (manuals, man **pages**, Info files) is in ...
www.tug.org/texlive/doc/texlive-en/live.**html** - Nov 30, 2005 - Similar pages

Treeview JavaScript **tree** menu
The frameless **tree** has additional **HTML**, unrelated to the **tree** itself and also ... For frameless **pages** with other elements besides the **tree**, the relative ...
www.**tree**menu.net/**tree**menu/instructions.asp - Similar pages

Portland, OR & Vancouver, WA - Our **Tree**
Family **Tree** Maker Home Page Make your own page, surf other user **pages** for free, ... Gedcom to **HTML** tools · GEDcom Analyzer - compare 2 gedcom files! ...
www.cemeteryinfo.net/misc/our**tree**.**html** - Similar pages

2 - Other OpenBSD Information Resources
These are found throughout the src **tree**. The man **pages** are found in the ... See report.**html** for more information on creating and submitting bug reports. ...
www.openbsd.org/faq/faq2.**html** - Similar pages

Result Page: **Previous** 1 2 3 4 5 6 7 8 9 10 **11** 12 13 14 15 16 17 18 19 20 **Next**

html tree pages Search

Search within results | Search Tips

Results for **"((xml<in>metadata) <and> (page<in>metadata))<and> (trees<in>met..."**
Your search matched **6** of **1263585** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

e-mail printer friendly

» Search Options

View Session History

New Search

Modify Search

((xml<in>metadata) <and> (page<in>metadata))<and> (trees<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Select Article Information

1. **Mining frequent rooted subtrees in XML data with Me-Tree**
Wansong Zhang; Daxin Liu; Jianpei Zhang;
Systems and Information Engineering Design Symposium, 2004. Proceedings of the 2004 IEEE
16 Apr 2004 Page(s):209 - 214
AbstractPlus | Full Text: PDF(397 KB) IEEE CNF

2. **Fuzzy rules for HTML transcoding**
Schaefer, R.; Dangberg, A.; Mueller, W.;
System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on
7-10 Jan 2002 Page(s):1385 - 1393
AbstractPlus | Full Text: PDF(415 KB) IEEE CNF

3. **Querying XML documents made easy: nearest concept queries**
Schmidt, A.; Kersten, M.; Windhouwer, M.;
Data Engineering, 2001. Proceedings. 17th International Conference on
2-6 April 2001 Page(s):321 - 329
Digital Object Identifier 10.1109/ICDE.2001.914844
AbstractPlus | Full Text: PDF(656 KB) IEEE CNF

4. **ERX: a data model for collections of XML documents**
Psaila, G.;
Knowledge and Data Engineering Exchange, 1999. (KDEX '99) Proceedings. 1999 Workshop on
7 Nov. 1999 Page(s):99 - 104
Digital Object Identifier 10.1109/KDEX.1999.836632
AbstractPlus | Full Text: PDF(56 KB) IEEE CNF

5. **Detecting changes in XML documents**
Cobena, G.; Abiteboul, S.; Marian, A.;
Data Engineering, 2002. Proceedings. 18th International Conference on
26 Feb.-1 March 2002 Page(s):41 - 52
Digital Object Identifier 10.1109/ICDE.2002.994696
AbstractPlus | Full Text: PDF(408 KB) IEEE CNF

6. **The visual specification of context**
Bruggemann-Klein, A.; Hermann, S.; Wood, D.;
Research and Technology Advances in Digital Libraries, 1999. ADL '99. Proceedings. IEEE Forum on
19-21 May 1999 Page(s):28 - 36
Digital Object Identifier 10.1109/ADL.1999.777688

AbstractPlus | Full Text: PDF(100 KB) IEEE CNF

Results for "((html<in>metadata) <and> (page<in>metadata))<and> (trees<in>me..."

e-mail printer friendly

Your search matched **14** of **1263585** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

((html<in>metadata) <and> (page<in>metadata))<and> (trees<in>metadata)

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Select Article Information

1. **Segmenting the Web document with document object model**
Jianli Luo; Jie Shen; Cuihua Xie;
Services Computing, 2004. (SCC 2004). Proceedings. 2004 IEEE International Conference on
15-18 Sept. 2004 Page(s):449 - 452
Digital Object Identifier 10.1109/SCC.2004.1358040
AbstractPlus | Full Text: PDF(293 KB) IEEE CNF

2. **Fuzzy rules for HTML transcoding**
Schaefer, R.; Dangberg, A.; Mueller, W.;
System Sciences, 2002. HICSS. Proceedings of the 35th Annual Hawaii International Conference on
7-10 Jan 2002 Page(s):1385 - 1393
AbstractPlus | Full Text: PDF(415 KB) IEEE CNF

3. **Applications of YK algorithm to the Internet transmission of Web-data: implementation issues and modifications**
Banerji, A.; En-hui Yang;
Data Compression Conference, 2000. Proceedings. DCC 2000
28-30 March 2000 Page(s):546
Digital Object Identifier 10.1109/DCC.2000.838193
AbstractPlus | Full Text: PDF(16 KB) IEEE CNF

4. **Hypermedia fusion-a document generator for the World Wide Web**
Caumanns, J.; Lenz, H.-J.;
Multimedia Computing and Systems, 1999. IEEE International Conference on
Volume 2, 7-11 June 1999 Page(s):892 - 896 vol.2
Digital Object Identifier 10.1109/MMCS.1999.778606
AbstractPlus | Full Text: PDF(408 KB) IEEE CNF

5. **VHDL2HYPER-a highly flexible hypertext generator for VHDL models**
Ecker, W.; Heuchling, M.; Mades, J.; Schneider, C.; Schneider, T.; Windisch, A.; Ke Yang; Zarabaldi, M.;
Fall VIUF Workshop, 1999.
4-6 Oct. 1999 Page(s):57 - 62
Digital Object Identifier 10.1109/VIUF.1999.801978
AbstractPlus | Full Text: PDF(2180 KB) IEEE CNF

6. **WIRE-a WWW-based information retrieval and extraction system**
Aggarwal, S.; Hung, F.; Weiyi Meng;
Database and Expert Systems Applications, 1998. Proceedings. Ninth International Workshop on
26-28 Aug. 1998 Page(s):887 - 892

Digital Object Identifier 10.1109/DEXA.1998.707510

AbstractPlus | Full Text: PDF(48 KB) IEEE CNF

7. **Working with remote VRML scenes through low-bandwidth connections**
Raposo, A.B.; Magalhaes, L.P.; Ricarte, I.L.M.;
Computer Graphics and Image Processing, 1997. Proceedings., X Brazilian Symposium on
14-17 Oct. 1997 Page(s):34 - 41
Digital Object Identifier 10.1109/SIGRA.1997.625145

AbstractPlus | Full Text: PDF(676 KB) IEEE CNF

8. **Web site optimization using page popularity**
Garofalakis, J.; Kappos, P.; Mourloukos, D.;
Internet Computing, IEEE
Volume 3, Issue 4, July-Aug. 1999 Page(s):22 - 29
Digital Object Identifier 10.1109/4236.780957

AbstractPlus | References | Full Text: PDF(980 KB) IEEE JNL

9. **WISE: A Visual Tool for Automatic Extraction of Objects from World Wide Web**
Yi-Wei Jan; Jyh-Jong Tsay; Bo-Liang Wu;
Web Intelligence, 2005. Proceedings. The 2005 IEEE/WIC/ACM International Conference on
19-22 Sept. 2005 Page(s):590 - 593
Digital Object Identifier 10.1109/WI.2005.167

AbstractPlus | Full Text: PDF(248 KB) IEEE CNF

10. **Web applications design and maintenance using symbolic model checking**
Di Sciascio, E.; Donini, F.M.; Mongiello, M.; Piscitelli, G.;
Software Maintenance and Reengineering, 2003. Proceedings. Seventh European Conference on
26-28 March 2003 Page(s):63 - 72
Digital Object Identifier 10.1109/CSMR.2003.1192411

AbstractPlus | Full Text: PDF(655 KB) IEEE CNF

11. **Robust pointing by XPath language: authoring support and empirical evaluation**
Abe, M.; Hori, M.;
Applications and the Internet, 2003. Proceedings. 2003 Symposium on
27-31 Jan. 2003 Page(s):156 - 165

AbstractPlus | Full Text: PDF(657 KB) IEEE CNF

12. **HTML page analysis based on visual cues**
Yudong Yang; HongJiang Zhang;
Document Analysis and Recognition, 2001. Proceedings. Sixth International Conference on
10-13 Sept. 2001 Page(s):859 - 864
Digital Object Identifier 10.1109/ICDAR.2001.953909

AbstractPlus | Full Text: PDF(640 KB) IEEE CNF

13. **A fast image segmentation algorithm for interactive video hotspot retrieval**
Tiecheng Liu; Kender, J.R.; Hjelsvold, R.; Pizano, A.;
Content-Based Access of Image and Video Libraries, 2001. (CBAIVL 2001). IEEE Workshop on
14 Dec. 2001 Page(s):3 - 8
Digital Object Identifier 10.1109/IVL.2001.990849

AbstractPlus | Full Text: PDF(742 KB) IEEE CNF

14. **The visual specification of context**
Bruggemann-Klein, A.; Hermann, S.; Wood, D.;
Research and Technology Advances in Digital Libraries, 1999. ADL '99. Proceedings. IEEE Forum on
19-21 May 1999 Page(s):28 - 36
Digital Object Identifier 10.1109/ADL.1999.777688

AbstractPlus | Full Text: PDF(100 KB) IEEE CNF

WEB SITE OPTIMIZATION USING PAGE POPULARITY

A link-editing algorithm based on relative page popularity can automatically revise a Web site's page structure to create a substantially more effective information scheme.

JOHN GAROFALAKIS, PANAGIOTIS KAPPOS, AND DIMITRIS MOURLOUKOS
University of Patras, Greece

An HTTP log file records every request that a server accepts, including information such as the date and time of the request, the client submitting it, the file requested, and so on.[1] This neat source of information establishes a basis for observations that can lead to improved overall performance for a given Web site. One of these observations, page popularity, is the subject of this article.

We examine how to define page popularity in a meaningful way and show how using page popularity to rearrange a Web site can lead to a substantially more accessible and more effective hypertext scheme. (In a similar vein, Golovchinsky has explored automatic methods of link construction based on feedback from users collected during browsing.[2])

The notion of log file analysis is not new (see the sidebar, "Web Resources for Improving Navigation" on page 24), but our approach uses an analysis of relative page popularity to automatically reorganize the pages of a Web site. We have developed pilot software to perform these functions and show here that the resulting improvement in navigation boosted page accesses in five different Web sites.

DEFINING PAGE POPULARITY

One easy way to estimate a page's popularity is to count the accesses to this page based exclusively on a given log file. (See http://www.w3.org/Daemon/User/Config/Logging.html for all the http connection properties written to the log file.) However, counting these *absolute accesses* (AAs) from the log file can be misleading. A page close to the home page (the server's initial page) will probably have more absolute accesses, because it stands on a path between the home page and target pages located deeper in the HTML "tree", where the home page is the root, and every hyperlink to another page constitutes a parent-child relationship. (Because a child page might have several links toward one of its ancestors, the term "tree" is not wholly felicitous.)

To better examine a page's popularity, we could take the following factors into consideration:

1089-7801/99/$10.00 ©1999 IEEE

- the depth of the page (how many steps it is from the home page), d;
- the number of pages at the same depth as the page being examined, n; and
- the number of references (hyperlinks) to this particular page from other pages of the server, r.

Let's assume there is a factor a that embraces all of these parameters. Then, we can coin a new term, *relative accesses* (RA), which we derive from the following equation:

$$\text{RA} = a * \text{AA} \qquad (1)$$

Defining the Coefficient *a*

Because page depth d can detract from the popularity of a page, we can reasonably assume that coefficient a must be proportional to d. Likewise, the number of pages at the same depth, n_d, should be proportional to a because the larger the number of choices, the greater the significance of selecting a specific page. Of course, references from other pages, r, generally bolster page popularity, so r must be in inverse proportion to coefficient a.

Recent Web servers can be configured to track down Web pages that have links to any of their pages and store the addresses in what is called the *referrer log*. Most Web browsers support this feature and thus send the reference (link) to a server's page simultaneously with the browser's HTTP request.

Let r be an estimate based on the number of references from a server's pages to a page on the same server. The relation between a_i, d_i, n_i, and r_i—the values of a, d, n, and r for page i—should have the following form:

$$a_i = F(d_i, n_i, 1/r_i), \; i = 1 \ldots K, \qquad (2)$$

where K is the number of Web pages and F is a specific function that can be defined to reflect a user's behavior—what pages the user visits, in what order, for how long, and so on—during navigation to a specific Web site.

Link-Editing Software

We have developed pilot software, Soala Version 2.0,[3] that reorganizes the links between pages according to criteria set by the Web server's administrator. In the following measurements, if the relative popularity of a given page exceeds the popularity of at least one ancestor page (a page with shorter distance to the home page, standing in its path), the software rearranges the links.

Soala can assess relative page popularity periodically (every week, for example). The most instructive of the metrics it calculates are

- AA—number of absolute accesses per page;
- RA—number of relative accesses per page;
- PT—mean page time, how much time a user spends on a specific page;
- UT—mean user time, how much time a user spends on the server every session; and
- NP—mean number of pages, how many different pages a user visits every session.

Soala reorganizes the links between pages according to criteria set by the Web server's administrator.

In our analyses, a new session begins when a user first accesses the Web site. All successive accesses belong to the same session provided that the interval between them does not exceed a given time threshold.

A simple version of our link-editing algorithm is "If any page has an RA higher than its parent, interchange it with its parent page. Repeat the previous step until every page has ancestors with higher RAs."

Case Study Definition

We applied this algorithm for a specific definition of factor a_i to show how an awkward organization of a Web server can discourage Internet users and how link editing (based on relative accesses) can remedy the problem.

To ensure that our results would be as accurate as possible, we allowed only authenticated users with valid usernames and passwords to access the Web server for this case study. The authentication of users is not crucial for the link-editing algorithm; we introduced this requirement for evaluation purposes only. Furthermore, there were no references to our site from outside our server, so the value of parameter r_i was accurate. We carefully selected the group of users, as well as the pages, to facilitate the interpretation of the results.

According to our earlier discussion, one possible way of specifying coefficient a_i is

$$a_i = c_1 * d_i + c_2 * n/r_i \qquad (3)$$

where c_1 and c_2 are constants whose values vary according to the structure of the Web site.

The definition of parameters c_1 and c_2 is definitely a challenging issue. Currently, we know that c_1 and c_2 can significantly affect the algorithm's behavior, but we are still researching what values will boost its

WEB RESOURCES FOR IMPROVING NAVIGATION

Our approach to improving Web site navigation automatically reorganizes pages on the basis of relative page popularity. This represents a different objective from other tools and, therefore, it is possible to combine our approach with other methods.

On the Market

There are many HTTP log file statistical analyzers on the market. For example, see

Analog Log Analyser • http://www.statslab.cam.ac.uk/~sret1/analog/
GwStat • http://dis.cs.umass.edu/stats/gwstat.html
WebTrends • http://www.webtrends.com

None of them, however, suggests any site rearrangement based on their statistical data.

There are also navigation tools that include guidelines on how to improve Web site navigation, usability, and appeal. For example, see

Usable Web, "Web Architecture" • http://usableweb.com/items/architecture.html
Microsoft, "Improving Web Site Usability and Appeal," • http://msdn.microsoft.com/workshop/management/planning/improvingsiteusa.asp
Web Developer's Virtual Library, "Navigation," • http://www.stars.com/Location/Navigation/

But these tools do not work automatically. On the contrary, their effectiveness depends almost entirely on the administrator's judgment and experience in Web design.

Web site promotion tools focus on how to attract more visitors to a specific site, either by achieving higher scores in search-engine queries—

SiteOwner Link Exchange Site • http://www.siteowner.com/
Submit It! Announcement Service • http://www.submit-it.com/
Cyberguide to Search Engine Promotion • http://www.rankthis.com/

—or by participating in Web site rings; see

WebRing • http://www.webring.org/

Although these tools may be valuable, they do nothing about structure enhancement, which is the focus of our approach.

In the Research Community

To our knowledge, the scientific literature includes no other published work on the analysis of log files according to page popularity. There is, however, some work in related fields. For example, Creech[1] proposes an author-oriented link management system to help authors ensure the consistency of their Web material whenever it is moved, deleted, or changed. This method focuses on informing the authors whenever their pages have incorrect links or trying to automatically repair broken links whenever possible.

Researchers have acknowledged the importance of the depth versus breadth in the arrangement of Web links—

K. Larson and M. Czerwinski, "Web Page Design: Implications of Memory, Structure, and Scent for Information Retrieval," *Proc. CHI 98: Human Factors in Computing Systems*, Addison-Wesley, Reading, Mass., 1998, pp. 25-32; available online at http://research.microsoft.com/users/marycz/chi981.htm.
P. Zaphiris and L. Mtei, "Depth vs Breadth in the Arrangement of Web Links," unpublished paper, available online at http://www.otal.umd.edu/SHORE/bs04/index.html.

—but no attempt has been made to connect this concept with user feedback as recorded in the log file.

In fact, good Web design is still an open research issue.[2,3] Our approach represents a new way to gain better overall site performance.

REFERENCES

1. M. Creech, "Author-Oriented Link Management," *Computer Networks and ISDN Systems*, Vol. 28, 1996, pp. 1015-1025.
2. R. Beasley and M. Waugh, "Cognitive Mapping Architectures and Hypermedia Disorientation: An Empirical Study," *J. Educational Multimedia and Hypermedia*, Vol. 4, No. 2/3, 1995, pp. 239-255.
3. A. Baylor, "Disorientation on the Web—Adventure or Distraction?" *Proc. WebNet '98, World Conference of the WWW, Internet, and Intranet*, Assoc. for the Advancement of Computing in Education (AACE), Charlottesville, Va., 1998, pp. 49-54.

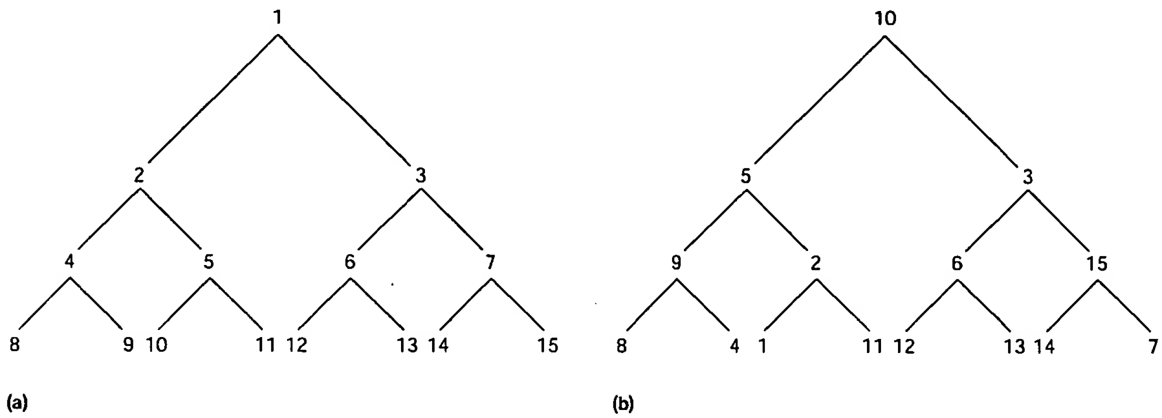

Figure 1. The link-editing algorithm reorganizes the pages in a Web site: (a) initial HTML structure and (b) revised HTML structure.

performance. For example, the higher we make the value of c_1, the more vigorously the algorithm promotes pages deeper in the HTML structure. In some cases (when the value of c_1 is extremely high), this leads to volatile page arrangements: after each execution of the link-editing algorithm, a new rearrangement occurs. This is not a sound case. Any robust algorithm should achieve a balanced page arrangement, which remains unaffected by further executions of the algorithm, if the relative accesses don't change significantly. If we assign the value 1 to parameters c_1 and c_2, we can rewrite equation 3 as

$$a_i = d_i + n/r_i$$

This is the case that we will examine.

First, from all PT and UT collected, we discarded time values that were too short or too long. A very short time represents the case of a user just clicking through a link to the next page. A very long time (one hour or more) represents the case of a user suspending navigation through the site for an external reason (an important e-mail or a meeting, for example) and returning much later to continue.

The site we evaluated comprised 15 pages (Page1, Page2, ... , Page15) organized into the binary tree structure depicted in Figure 1a. (Our algorithm also works on graph organizations; see the sidebar, "Link Editing for Graph Sites.") Page1, the initial home page, has two children, Page2 and Page3. For every n from 1 through 7, Page$n$ has two children: Page$2n$ and Page$(2n+1)$. There is only one reference to each page (from its parent, with the exception of the home page); therefore parameter r has a value of 1 in every case. Our algorithm maintains the binary structure in its revision, depicted in Figure 1b. What changes is the position of the pages within the structure. For instance, at the end of the link-editing procedure, Page2 occupies a position at a deeper level of the HTML tree than it did initially.

Table 1 (next page) summarizes the accesses to our system prior to link editing. For example, Page2 has a mean page time of 27 seconds, 97 absolute accesses, and 291 relative accesses (RA = a * AA, where $a = d + n/r = 1 + 2/1 = 3$).

As we can infer from Table 1, the initial HTML structure is rather inapt. It undervalues pages such as Page10, which exhibits significantly higher RA than other pages located closer to the home page. Table 2 summarizes the measurements based on the revised binary tree.

As we can easily see from Table 2, the new tree organization results in pages with a substantially different AA. The new sum of AAs

SUM(AA) = AA(Page1) + AA(Page2) + ...
+AA(Page15)

exhibits a significant increase: the old SUM(AA) was 726; the new SUM(AA) is 866. The mean user time, UT, has also increased—from 112 seconds to 146 seconds—whereas PT fluctuates. Furthermore, the new page arrangement boosts the mean number of pages per user (NP) from 2.04 to 2.67, an increase similar to that for UT.

Even though these measurements are statistical, we can draw significant inferences. First, even a simple approach to link editing, based on relative page accesses, can significantly affect a user's behavior towards a particular Web site. Comparing Table 1 to Table 2 (next page), we can infer that we

Table 1. Measurement of accesses prior to link editing (Figure 1a).

Page	PT (seconds)	AA	RA
1	41	227	227
2	27	97	291
3	153	117	351
4	29	30	180
5	52	60	360
6	17	28	168
7	61	43	258
8	32	8	88
9	47	17	187
10	131	41	451
11	29	7	77
12	57	14	154
13	33	9	99
14	11	3	33
15	101	25	275

Table 2. Measurement of accesses after link editing; revised HTML structure (Figure 1b).

Page	PT (seconds)	AA	RA
1	43	11	121
2	28	37	222
3	151	160	480
4	31	8	88
5	49	103	309
6	17	43	258
7	63	16	176
8	30	12	132
9	45	47	282
10	121	326	326
11	32	13	143
12	62	21	231
13	35	12	132
14	13	2	22
15	97	55	330

increased the number of accesses to our site by 19 percent without changing the content of the pages. The 31 percent higher NP and UT indicate that the new organization seems to be more convenient and more attractive to potential users.

Another important aspect of this reorganization is that it can be automated: using a simple link-editing algorithm, we put the Web site structure in a more balanced and organized state (from a relative-access point of view).

APPLICATION RESULTS

The platform requirements of the Soala software are moderate. Any Pentium-based personal computer with 16 Mbytes of RAM is sufficient for the Soala program to run and calculate page popularities. The Web server we used in our research was the Apache 2.5.1 compiled for Solaris 2.6. Because the algorithm is independent of the Web server software, any Web server could serve the original or modified Web pages.

We applied our algorithm to several Web sites with totally different page structures (each created on an ad hoc basis) and calculated the percentage of the improvement in page accesses, shown in Table 3. Clearly, the more inapt an HTML structure, the more notably our link-editing algorithm improved it. To test how specific characteristics, such as contextual coherence among the pages, affected our algorithm's efficiency, we created sites that corresponded closely with the given requirements.

To further test the Soala software, we developed a Web site called Theleis. The objective of this new service is to help Greek Internet users find other services, products, and information that might be of interest to them. We built a series of 70 Web pages with variable information in their links and included two user prizes. So that we could take full advantage of Soala's capabilities, we left the architecture (how Web pages were linked, how many levels there should be, which page should go in

Table 3. Improvement in Web page accesses through link editing.

Site description	Number of nodes	AA improvement (%)
Inapt binary-tree page arrangement. Every page can be freely interchanged with its immediate relatives.	31	21
Inapt binary-tree page arrangement. Every page can be freely interchanged with its immediate relatives.	15	19
Graph HTML structure partially organized. Only the initial home page defined as a stable node.	31	12
Well-organized site. Initial home page and its children defined as stable nodes.	15	4

which level, and so on) rather open ended.

We did use some logical restrictions for the initial structure of Theleis, and we made some arbitrary decisions about what the site should look like. Then, we left the site to be accessed for 15 days. Next, we used the Soala link-editing algorithm to revise the initial HTML structure, and we retested the accessibility for the same time period (15 days, the same days of the week). As Table 4 shows, the results were encouraging. There was a 14 percent increase in absolute accesses. The 11 percent increase in the average page time shows that the site has become more attractive to users.

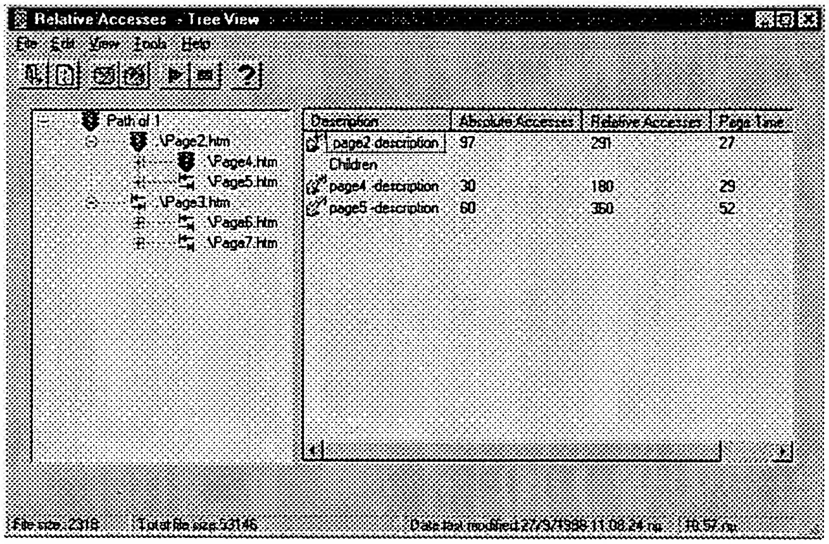

Figure 2. The link-editing interface for the Soala software. With a simple mouse click, the user can define whether a node—or a link between two nodes—is stable. The interface uses four icons to indicate the current state of each node and link. In addition, the GUI gives the user a visual representation of the HTML structure. Clicking on a specific node gives the user detailed information about and a description of that node and its immediate relatives.

The links among pages constitute a "grafo" (a graph), and the access to the site is anonymous (no login name or password is required). Every page includes a link to the home page in addition to the alternative links a visitor might choose. Furthermore, to tempt the user into staying longer, we added some gift pages hidden in a deep level of the structure, toward the leaves.

Table 4. Theleis site: access measurements before and after link editing.

Site	Number of absolute accesses	Average page time (in seconds)
Initial	13,305	6.5
Revised	15,168	7.2

LIMITATIONS OF OUR APPROACH

There are cases in which an HTML rearrangement is not feasible, because the structure of the Web server follows the dictates of logical restrictions. For example, take the case of a site used as a lyrics archive. The initial page would be a table of contents (A–Z, where A leads to all artists whose names start with A, and so on); the leaves of the HTML tree would be the songs of each artist. Even if a particular song has substantially more accesses than the others, it cannot become the root of the HTML tree, because this would make navigation to the rest of the lyrics unjustifiably problematic.

So Web page content is also important, and the administrator should consider it in page rearrangement. Soala 2.0 tackles such problems by enabling the administrator to define stable nodes and stable connections, as illustrated in Figure 2. If the administrator defines a connection as stable, the connected nodes cannot be interchanged. The same applies to a stable node; it cannot be interchanged with any of its parent or child nodes.

Another side effect of any link-editing algorithm is that changing the positions of the HTML pages can render links from external sites (such as search engines, subject catalogs, and so on) out of date. Such broken links can be annoying for Net surfers, and the administrator should bear this effect in mind, especially if the link-editing algorithm is executed frequently and results in the replacement of many useful pages.

D. Ingham and colleagues have proposed an object-oriented approach to defeat broken links,[4] and F. Kappe has proposed a server update protocol to propagate update information.[5] There is a much easier method, however, that demands little effort from the Web administrator. During the link-editing procedure, the Soala software can map

LINK EDITING FOR GRAPH SITES

In our case study, we assumed a binary-tree structure for evaluation purposes only. Our link-editing algorithm works just as well reorganizing a graph site, which permits all kinds of links—nodes with links to a sibling node, an ancestor, and so on. Figure A shows the reorganization of a graph site.

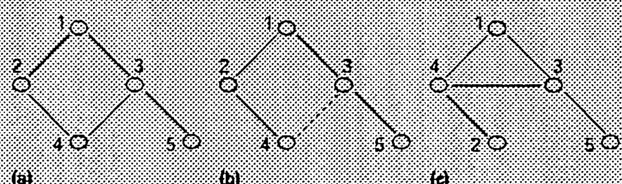

Figure A. Reorganization of a graph Web site. (a) Before we apply our link-editing algorithm, we assign each node a unique id number. Node 4 (the node with id = 4) is linked with both node 2 and node 3, but node 2 is regarded as its parent, because it has a lower id than node 3. (b) During execution, the algorithm ignores the link between node 4 and node 3. (c) When the rearrangement is complete, the algorithm reestablishes the link between nodes 4 and 3.

the URLs of the pages being replaced and their new URLs and put them into a database (either a relational database or a simple file accessible from the Web server). Then, the administrator can specially configure the error handler to check this database when a user pursues a nonupdated external link. (See http://www.apache.org/docs-1.2/custom-error.html for information on how to configure the Apache Web server to handle such error conditions.) Instead of returning the "404 Not found" message, the error handler determines whether the outdated link has been replaced by another one during link editing. If this page exists in the database, a message informs the user of its new URL and automatically redirects the browser to point to the new location of the page. If the file does not exist, the browser displays an error message. In this way, even if the link-editing algorithm has modified all the Web server's pages and changed their URLs, there will be no broken links, even on references from search engines.

To reduce the additional load on the Web server caused by execution of the error script handler, we advise the administrator to resubmit the home page URL to the search engine so it can update its links. The update will be available to all network users after the search engine's reorganization interval, usually a month.

CONCLUSION

An awkward arrangement of an HTML tree can discourage a potential Internet user from visiting and staying at a specific Web site. We have shown how a link-editing algorithm can automatically fix a poor organization by calculating each page's relative popularity. Even a simple approach using relative page popularity—the combination of our equations 1 and 2—leads to a substantially enhanced scheme.

In this article, we considered only cases where the objective is to make it easier for a user to find the requested data: the faster the access, the better the organization of the Web server. But what if a Web site contains commercial material—advertisements and so on? In this case, the objective may be different: the best organization may be one that achieves the highest AA for those pages with a commercial interest. In these cases, we can associate a weight factor w with each page and modify the link-editing algorithm so that it promotes pages with a higher w more vigorously. We have developed a version of our link-editing algorithm to handle this association. A beta version, still under evaluation, has returned encouraging results.

Surely there are many different approaches to rearranging links between pages and many different ways of perceiving the notion of popularity. For example, we could give coefficient a a more complicated definition than that given in equation 2, or we could rearrange the initial page structure using a more sophisticated method than our link-editing algorithm. Nevertheless, the approach we presented in this article should not be underrated; besides being easy to analyze, it has also proved adequate in most cases, considering the percentage increase in page accesses. A link-editing algorithm that could add new links and discard others (in addition to interchanging pages) would probably demonstrate a better enhancement.

Our team is still in the process of identifying several new variables that affect Web page popularity. We will make our new results available as soon as they are extracted. ■

ACKNOWLEDGMENTS

We presented an early version of this article at WebNet 98: World Conference of the WWW, Internet, and Intranet, Nov. 1998, Orlando, Florida; it appears in the proceedings of that conference, published by the Assoc. for the Advancement of Computing in Education (AACE), Charlottesville, Va., USA, pp. 324-329.

REFERENCES

1. HTTP Request Fields (RFC822 header lines), available online at http://www.w3.org/Protocols/HTTP/HTRQ_Headers.html.
2. G. Golovchinsky, "What the Query Told the Link: The Integration of Hypertext and Information Retrieval," *Proc. 8th ACM Conf. Hypertext*, ACM Press, New York, 1997, pp. 67-74.
3. *SOALA*, Tech. Report CTI-TR98.3.16, Computer Technology Inst., Patras, Greece, 1998.
4. D. Ingham et al., "W3Objects: Bringing Object-Oriented Technology to the Web," *The Web J.*, Vol. 1, No. 1, pp. 89-105.
5. F. Kappe, "A Scalable Architecture for Maintaining Referential Integrity in Distributed Information Systems," *J. Universal Computer Science*, Vol. 1, No. 2, Feb. 1995, pp. 84-104.

John Garofalakis is an assistant professor in the Department of Computer Engineering and Informatics and head of a research unit at the Computer Technology Institute, Patras, Greece. His research interests include performance evaluation of computer systems, distributed systems and algorithms, and Internet technologies and applications. Garofalakis received his PhD from the Department of Computer Engineering and Informatics, University of Patras, and his diploma in electrical engineering from the National Technical University of Athens.

Panagiotis Kappos is a researcher at the Research Unit 5 of the Computer Technology Institute. His research interests include Internet and Web technologies, electronic commerce, and network security. He was a referee for the ACM SIGCHI 99 Conference. Kappos received his diploma from the Department of Computer Engineering and Informatics, University of Patras. He is currently working toward a PhD from the same department.

Dimitris Mourloukos is a system administrator and researcher in the Wire Communications Laboratory, Department of Electrical and Computer Engineering, University of Patras. Among his research interests are text-to-speech synthesis and sophisticated WWW site development strategies. Mourloukos received his diploma from the Department of Computer Engineering and Informatics, University of Patras.

Readers can contact Kappos at the Department of Computer Engineering and Informatics, University of Patras, 26500 Patras, Greece; kappos@cti.gr.

Access this document

Full Text: PDF (980 KB)

Download this citation

Choose Citation

Download EndNote,ProCite,RefMan

» Learn More

Rights & Permissions

Request Permissions RIGHTSLINK

» Learn More

Web site optimization using page popularity

Garofalakis, J. Kappos, P. Mourloukos, D.
Dept. of Comput. Eng. & Inf., Patras Univ. , Greece;

This paper appears in: **Internet Computing, IEEE**
Publication Date: July-Aug. 1999
Volume: 3 , Issue: 4
On page(s): 22 - 29
ISSN: 1089-7801
CODEN: IICOFX
INSPEC Accession Number:6327292
Digital Object Identifier: 10.1109/4236.780957
Posted online: 2002-08-06 22:38:11.0

Abstract
Awkward arrangement of documents in an **HTML tree** can discourage users from staying at a Web site. The authors have developed an algorithm for dynamically altering the organization of **pages** at sites where the main design objective is to give users fast access to requested data. The algorithm reads information from th HTTP log file and computes the relative popularity of **pages** within the site. Based on popularity (defined as a relationship between number of accesses, time spent, and location of the **page**), the hierarchical relationship between **pages** are rearranged to maximize accessibility for popular **pages**

Index Terms
Inspec
Controlled Indexing
Internet hypermedia markup languages information resources user interfaces

Non-controlled Indexing
HTML HTTP log file Web site optimization data access hierarchical relationship **page** organization **page** popularity
Author Keywords
Not Available

References

1 HTTP Request Fields (RFC822 header lines), available online at http://www.w3.org/Protocols/HTTP/HTRQ_Headers.html.

2 G. Golovchinsky, "What the Query Told the Link: The Integration of Hypertext and Information Retrieval," *Proc. 8th ACM Conf. Hypertext*, ACM Press, New York, 1997, pp. 67-74.
[Buy Via Ask*IEEE]

3 *SOALA*, Tech. Report CTI-TR98.3.16, Computer Technology Inst., Patras, Greece, 1998.

4 D. Ingham et al., "W3Objects: Bringing Object-Oriented Technology to the Web," *The Web J.*, Vol. 1, No. pp. 89-105.
[Buy Via Ask*IEEE]

5 F. Kappe, "A Scalable Architecture for Maintaining Referential Integrity in Distributed Information System *J. Universal Computer Science*, Vol. 1, No. 2, Feb. 1995, pp. 84-104.
[Buy Via Ask*IEEE]

Sidebar A References

1 M. Creech, "Author-Oriented Link Management," *Computer Networks and ISDN Systems*, Vol. 28, 1996, pp. 1015-1025.
[Buy Via Ask*IEEE]

2 R. Beasley and M. Waugh, "Cognitive Mapping Architectures and Hypermedia Disorientation: An Empiric Study," *J. Educational Multimedia and Hypermedia*, Vol. 4, No. 2/3, 1995, pp. 239-255.

[Buy Via Ask*IEEE]

3 A. Baylor, "Disorientation on the Web—Adventure or Distraction?" *Proc. WebNet '98: World Conf. of the WWW, Internet, and Intranet*, Assoc. for the Advancement of Computing in Education (AACE), Charlottesville, Va., 1998, pp. 49-54.
[Buy Via Ask*IEEE]

Citing Documents

1 Automatic construction of online catalog topologies, Wing-Kin Sung; Yang, D.; Siu-Ming Yiu; Cheung, D.W.; Wai-Shing Ho; Tak-Wah Lam
Systems, Man and Cybernetics, Part C, IEEE Transactions on
On page(s): 382- 391, Volume: 32, Issue: 4, Nov 2002
Abstract | Full Text: PDF (418)

Refine Search

Search Results -

Terms	Documents
L6 and trees	3

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Friday, December 02, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L7	L6 and trees	3	L7
L6	L5 and pages	14	L6
L5	L4 and ("hypertext markup language" or "html")	14	L5
L4	L3 and (crawl$ or spider$)	27	L4
L3	L1 and (suction or buy$ and sell$ or bidd$ or cross-sell$) near sites	254	L3
L2	L1 and (suction or buy$ and sell$ or bidd$) near site	254	L2
L1	(internet or www or network or web)	1724289	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Terms	Documents
"html" and "xml" near2 page near trees	4

Database: US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Friday, December 02, 2005 Printable Copy Create Case

Set Name side by side	**Query**	**Hit Count**	**Set Name** result set
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L10	"html" and "xml" near2 page near trees	4	L10
	DB=USPT; PLUR=YES; OP=OR		
L9	'6356899'.pn.	1	L9
L8	'6356899'.pn.	1	L8
L7	'6526424'.pn.	1	L7
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L6	L5 and collect$ near2 (data or information)	30	L6
L5	L4 and ("xml" or "extensible markup language")	61	L5
L4	L3 and (page with trees or page near trees or page adj trees)	128	L4
L3	L2 and ("html" or "hypertext markup language")	1714	L3
L2	(crawl$ or spider$)	102154	L2
L1	(crawl$ or spider$) near2 (html or "hypertext markup language") near2 trees	0	L1

END OF SEARCH HISTORY

Freeform Search

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term:

Display: 10 **Documents in Display Format:** - **Starting with Number** 1

Generate: ○ Hit List ◉ Hit Count ○ Side by Side ○ Image

Search Clear Interrupt

Search History

DATE: Friday, December 02, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=USPT; PLUR=YES; OP=OR		
L46	'5897622'.pn.	1	L46
L45	'5897622'.pn.	1	L45
L44	'6151624'.pn.	1	L44
	DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L43	L42 and (enterprises or organizations or organisations)	73	L43
L42	L41 and (class or category) near2 (items or merchandise or products)	119	L42
L41	l1 and 705/27	403	L41
L40	l2 and 705/27	12	L40
L39	l3 and 705/27	7	L39
	DB=USPT; PLUR=YES; OP=OR		
L38	'6101483'.pn.	1	L38
L37	'6101483'.pn.	1	L37
L36	'5978773'.pn.	1	L36
L35	'5978773'.pn.	1	L35
L34	'5950173'.pn.	1	L34
L33	'5950173'.pn.	1	L33
L32	'6401085'.pn.	1	L32

DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L31	15 and 705/27	7	L31
L30	5305200.pn.	2	L30
L29	5671279.pn.	2	L29
DB=USPT; PLUR=YES; OP=OR			
L28	(5497317 | 4677552 | 5267148 | 4412287 | 4674044 | 5297032 | 5136501 | 4334270 | 5710889 | 5101353 | 5864827 | 6012046 | 5924083 | 5220500 | 5950177 | 5038284 | 5689652 | 5126936 | 5132899 | 5270922 | 5845266)![PN]	21	L28
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L27	('6278982')[ABPN1,NRPN,PN,TBAN,WKU]	2	L27
L26	6278982.pn.	2	L26
DB=USPT; PLUR=YES; OP=OR			
L25	(5317729 | 5907837 | 5455407 | 5970476 | 5737739 | 5949876 | 6151601 | 6067559 | 6157915 | 5263129 | 5842196 | 6119229 | 5862346 | 6078924 | 6085220 | 5311424 | 6182091 | 5273434)![PN]	18	L25
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L24	('6424979')[ABPN1,NRPN,PN,TBAN,WKU]	2	L24
L23	6424979.pn.	2	L23
DB=USPT; PLUR=YES; OP=OR			
L22	'5855008'.pn.	1	L22
L21	'5855008'.pn.	1	L21
L20	'6157947'.pn.	1	L20
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L19	6687710.pn.	2	L19
L18	6264104.pn.	2	L18
L17	6032129.pn.	2	L17
L16	5905975.pn.	2	L16
L15	5319542.pn.	2	L15
L14	4984155.pn.	2	L14
DB=USPT; PLUR=YES; OP=OR			
L13	'4992940'.pn.	1	L13
L12	'4992940'.pn.	1	L12
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L11	6072481.pn.	2	L11
L10	6035283.pn.	2	L10
L9	5794219.pn.	2	L9
DB=USPT; PLUR=YES; OP=OR			
L8	'5191613'.pn.	1	L8
L7	'5794219'.pn.	1	L7
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L6	L5 and (host with interface or host near interface or host adj interface)	5	L6
L5	L4 and (enterprises or organizations or companies)	134	L5
L4	L3 and exchange	135	L4
L3	L2 and (class or category) near2 (items or products or merchandise)	146	L3
L2	L1 and aggregat$ near information	342	L2

L1	(e-commerce or e-shopping or e-shopp$)	12104	L1

END OF SEARCH HISTORY